

Name:.....

ID No. ....

1. For each of the given program segments, read the code and write the output in the space provided below each program [Note: *Don't execute these programs on a computer.*] Use the following class definition:

```
1 // Definition of Account class.
2 class Account
3 {
4 public:
5     Account( int ); // constructor initializes balance
6     void credit( int ); // add an amount to the account balance
7     int getBalance(); // return the account balance
8 private:
9     int balance; // data member that stores the balance
10 }; // end class Account
```

```
1 // Member-function definitions for class Account.
2 #include <iostream>
3 using namespace std;
4
5 #include "Account.h" // include definition of class Account
6
7 // Account constructor initializes data member balance
8 Account::Account( int initialBalance )
9 {
10     balance = 0; // assume that the balance begins at 0
11
12     // if initialBalance is greater than 0, set this value as the
13     // balance of the Account; otherwise, balance remains 0
14     if ( initialBalance > 0 )
15         balance = initialBalance;
16
17     // if initialBalance is negative, print error message
18     if ( initialBalance < 0 )
19         cout << "Error: Initial balance cannot be negative.\n" << endl;
20 } // end Account constructor
21
22 // credit (add) an amount to the account balance
23 void Account::credit( int amount )
24 {
25     balance = balance + amount; // add amount to balance
26 } // end function credit
27
28 // return the account balance
29 int Account::getBalance()
30 {
31     return balance; // gives the value of balance to the calling function
32 } // end function getBalance
```

a. What is the output of the following main function?

```
1 int main()
2 {
3     Account account1( 3550 );
4
5     cout << "account1 balance: $" << account1.getBalance() << endl;
6 } // end main
```

*Your answer:*

b. What is the output of the following main function?

```
1 int main()
2 {
3     Account account1( -2017 );
4
5     cout << "account1 balance: $" << account1.getBalance() << endl;
6 } // end main
```

*Your answer:*

c. What is the output of the following main function?

```
1 int main()
2 {
3     Account account1( 1533 );
4
5     cout << "account1 balance: $" << account1.getBalance() << endl;
6     cout << "adding $253 to account1 balance" << endl;
7
8     account1.credit( 253 );
9     cout << "account1 balance: $" << account1.getBalance() << endl;
10 } // end main
```

*Your answer:*

d. What is the output of the following main function?

```
1  int main()
2  {
3      Account account1( 2770 );
4
5      cout << "account1 balance: $" << account1.getBalance() << endl;
6      cout << "adding $375 to account1 balance" << endl;
7
8      account1.credit( 375 );
9      cout << "account1 balance: $" << account1.getBalance() << endl;
10 } // end main
```

*Your answer:*

e. What is the output of the following main function?

```
1  int main()
2  {
3      Account account1( 799 );
4
5      cout << "account1 balance: $" << account1.getBalance() << endl;
6      cout << "adding -$114 to account1 balance" << endl;
7
8      account1.credit( -114 );
9      cout << "account1 balance: $" << account1.getBalance() << endl;
10 } // end main
```

*Your answer:*

2. For each of the given program segments, determine if there is an error in the code. If there is an error, circle the error and write the corrected code in the space provided after each problem. If the code doesn't contain error, write "no error." Use the following class definition:

```
1 // Definition of GradeBook class that stores the course name.
2 #include <string> // program uses C++ standard string class
3 using namespace std;
4
5 // GradeBook class definition
6 class GradeBook
7 {
8 public:
9     // constructor initializes course name
10    GradeBook( string );
11    void setCourseName( string ); // function to set the course name
12    string getCourseName(); // function to retrieve the course name
13    void displayMessage(); // display welcome message and course name
14 private:
15    string courseName; // course name for this GradeBook
16 }; // end class GradeBook
```

---

```
1 // Member-function definitions for class GradeBook.
2 #include <iostream>
3 using namespace std;
4
5 // include definition of class GradeBook from GradeBook.h
6 #include "GradeBook.h"
7
8 // constructor initializes courseName
9 // with string supplied as argument
10 GradeBook::GradeBook( string course )
11 {
12     setCourseName( course ); // initializes courseName
13 } // end GradeBook constructor
14
15 // function to set the course name
16 void GradeBook::setCourseName( string name )
17 {
18     courseName = name; // store the course name
19 } // end function setCourseName
20
21 // function to retrieve the course name
22 string GradeBook::getCourseName()
23 {
24     return courseName;
25 } // end function getCourseName
26
27 // display a welcome message and the course name
28 void GradeBook::displayMessage()
29 {
30     // display a welcome message containing the course name
31     cout << "Welcome to the grade book for\n" << getCourseName() << "!"
32         << endl;
33 } // end function displayMessage
```

a . The following code segment should create a new GradeBook object:

```
1 Gradebook gradeBook( "Introduction to C++", 25 );
```

*Your answer:*

b . The following code segment should set the course name for the gradeBook object:

```
1 setCourseName( gradeBook, "Advanced C++" );
```

*Your answer:*

c . The following code segment should ask the user to input a course name. That name should then be set as the course name of your gradeBook.

```
1 cout << "Please enter the course name:" << endl;  
2 cin.getline( inputName );  
3  
4 gradeBook.setCourseName();
```

*Your answer:*

d . The following code segment should output gradeBook's current course name:

```
1 cout << "The grade book's course name is: " << gradeBook.courseName << endl;
```

*Your answer:*